Reasoning about plans for effective communication of decision support

Reasoning about plans for effective communication of decision support*

Abigail S. Gertner Bonnie L. Webber
Department of Computer & Information Science
University of Pennsylvania
Philadelphia PA 19104-6389
(agertner,bonnie)@linc.cis.upenn.edu

1 Introduction

Successful communication of information is essential for decision-support system to be effective in influencing patient outcomes [2]. This is especially true in the case of real-time systems that are intended to interpret situations and respond quickly to alter a physician's course of action. As a mode of communication, *critiquing* was proposed as an alternative to traditional advising systems in the early 80's [7]. Miller argued convincingly for the advantages of a user-focused, reactive approach to information delivery, but as yet no critiquing systems appear on a current list of computer systems in routine clinical use (http://www-uk.hpl.hp.com/people/ewc/list.html).

In general, medical critiquing systems have taken one of two approaches: (1) compare the therapy plans of the physician to the recommendations of the system and comment on the differences [7, 5], or (2) examine the physician's plan for pre-defined constraint violations and comment if they are found [11]. Both approaches are essentially limited to a superficial analysis of what actions appear in the physicians plan, not why they may be there. As such, they are overly critical because they are unable to take account of different planning strategies that physicians may exhibit in addressing the goals at hand. This tendency to be over-critical may in part account for the absence of critiquing systems from routine clinical use.

The fact that ordinary interaction and communication takes account of the goals underlying a person's actions was first observed in work on cooperative response generation in dialogue systems [1]. These studies observed that the respondent to a question would often provide relevant information that was not explicitly requested or correct perceived mistakes in the questioner's plan. Such behavior, it was argued, followed from the respondent's inferring the plans that motivated questions.

The same argument can be made in the case of critiquing: producing an appropriate critique of an agent's intended actions requires an understanding of *why* the agent is doing them. As Shahar and Musen [8] point out, by inferring the goals underlying physician actions a system can be more accommodating by accepting alternative ways of addressing goals. As long as the physician is pursuing a goal that is acceptable to the system, it may not be necessary to critique her behavior.

We have identified two additional factors motivating plan recognition in support of critiquing:

Explanation Teach and Shortliffe's work [9] indicates that a key factor in the acceptability of a decision-support system is its ability to explain its reasoning. Critiquing can exploit plan recognition to provide an explanation that includes the goal the physician is pursuing by doing an action, and possibly why that goal is not justified.

Proposing alternatives Having access to the goal(s) underlying a physician's action can allow the system to see the action as an alternative to what it would recommend for addressing that goal. If its recommendation has advantages in cost, speed, non-invasiveness, etc., it can be presented to the physician in this way, while supporting the physicians' original intentions.

2 Recognizing trauma management plans

As part of our work on TraumAID, a system for providing real-time quality assurance during the initial definitive management of multiple trauma [12], we have developed a critiquing interface, TraumaTIQ, that reasons explicitly about the physician's

*This work was supported by the National Library of Medicine under grant R01 LM05217-01.

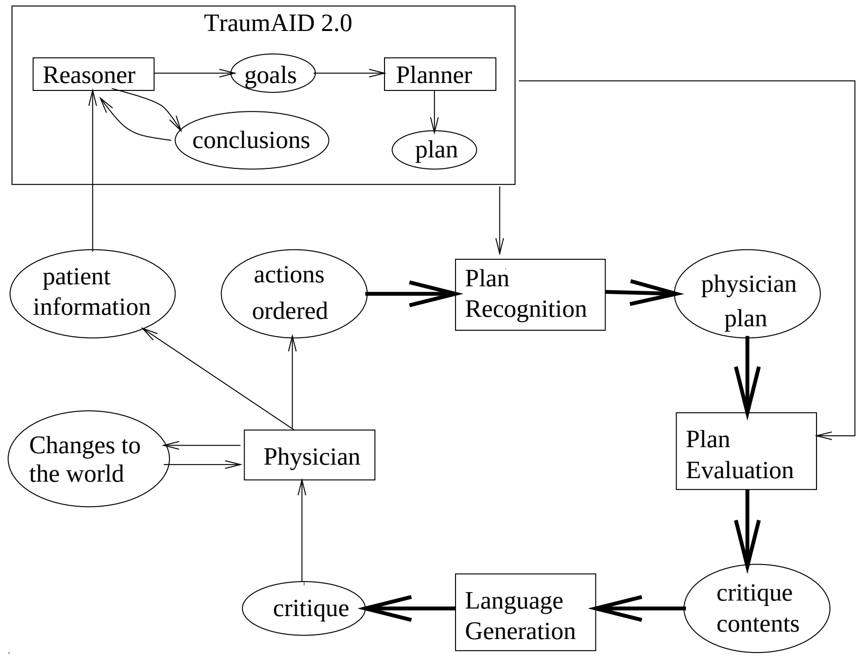

Figure 1: The TraumaTIQ module

goals and plans in order to critique her orders appropriately. As shown in Figure 1, TraumaTIQ functions in a repeating loop of plan recognition, plan evaluation, and critique generation that is triggered whenever new patient information or orders for action are entered into the system. It interfaces with the TraumAID system, which produces and updates its set of relevant goals and its recommended management plan on the basis of current patient information.

Trauma management is characterized by a heavy demand on the physician's attention due to the need to pursue multiple concurrent goals, often under critical time constraints. It is therefore necessary to abandon many of the assumptions that have typically been adopted by plan recognition systems, including the assumption that observed plans are correct (containing no actions that will not achieve the goals), that the agent pursues one top-level goal at a time, and that the agent is intentionally acting in such a way as to have her plan(s) recognized. Trauma management plans are made even more difficult to understand by the fact that single actions may have multiple purposes in the same plan.

Critiquing in support of trauma management also requires that plans be recognized incrementally, *during* patient management. Therefore, the plan recognition algorithm must take into account that at any given time the physician's plan is only partially specified.

Our approach to plan recognition under these circumstances exploits TraumAID's knowledge of the *context* in which plans are being developed. We make the assumption that physicians with some training and experience in the domain are likely to pursue goals that are *relevant* in the current situation. This leads to a policy of giving the physician the "benefit of the doubt," assuming, when at all possible, that her actions are being done for appropriate reasons.

Several researchers have pointed out the advantages of using contextual knowledge and basic domain principles to guide the search for an explanatory plan [4, 6]. The basic idea behind these approaches to plan recognition is that the plan recognizer can use its knowledge of what actions are appropriate to take in the current situation to reduce ambiguities in interpreting observed actions.

Plan recognition in TraumaTIQ takes advantage of three types of contextual information that influence the likelihood that the physician is pursuing a certain goal:

- **The likely goals to be pursued in the current situation:** Given TraumAID's current information about the state of the patient, TraumaTIQ is able to make certain inferences about what goals are more or less *relevant* to pursue.
- **The physician's actions:** The more evidence

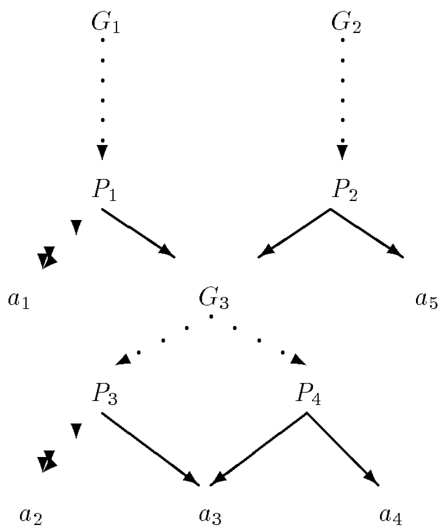

Figure 2: An example plan graph. Dotted arrows indicate disjunctive goal-procedure mappings, solid arrows indicate conjunctive procedure-action mappings

TraumaTIQ has that the physician is performing a procedure, the more likely it is that she is actually performing it.

- **The likelihood of procedures being used to pursue a goal:** While it may be possible to pursue a goal in a number of ways, some of them may be quite uncommon. This is reflected in the preference ordering for procedures in TraumAID's knowledge base.

3 The Plan Recognition algorithm

The task of the plan recognizer is to build incrementally a model of the physician's plan based on the actions being ordered. Following the assumptions given above, TraumaTIQ's plan recognizer prefers to explain the physician's actions in terms of goals (and procedures) that TraumAID currently considers *relevant* to the case.

A formal description of the plan recognition algorithm appears in Figure 3. Basically, it works as follows: It first enumerates the set of possible *explanations* for all actions that have been ordered. Each explanation consists of a path in the plan graph from the ordered action to a *procedure* in which the action plays a part, back to a top level *goal*. The path may pass through a series of sub-goals and procedures before reaching a top level goal. Since the same goal may be addressed by more than one procedure, it is possible for an action to be explained by a goal in the context of two different procedures. For example, in Figure 2, action a_3 is explained by goal G_3 through both procedures P_3 and P_4.

The possible explanations are evaluated in two phases. The first phase considers the *goals* in the explanations. These are sorted according to their *relevance* in the current situation. The plan recognizer categorizes potential explanatory goals according to a 4-level scale of relevance:

1. Relevant goals: goals that are in TraumAID's set of goals to be pursued.

2. Potentially relevant goals: goals that are part of a currently active *diagnostic strategy*. Diagnostic strategies are represented implicitly in TraumAID's knowledge base. They comprise chains of goals each of which, given the appropriate result, leads to the formation of the next goal in the strategy. So, for example, if the goal of diagnosing a fractured rib is currently relevant, then the goal of treating a fractured rib is potentially relevant, depending on the result of the diagnostic test.

3. Previously relevant goals: goals that were once relevant but are no longer relevant, either because they have been addressed or because some additional evidence has ruled them out.

4. Irrelevant goals: all other goals are classified as *irrelevant*.

The assumption underlying this phase of plan recognition is that the higher a goal is on this scale, the more likely the physician is to be pursuing it. The most relevant ones are selected as candidate explanations for the orders.

The likelihood that a goal is being pursued depends not just on its relevance, but also on the likelihood that the ordered actions would be used to address it. Therefore, goals that are potentially or previously relevant are not accepted as explanations unless the ordered actions play a role in the *most preferred* procedure for addressing those goals.

As others have pointed out [10], depending on the reason for doing plan recognition, it is not always necessary to infer a unique goal or goals for every action. For the purpose of critiquing, we do not want to spend time interpreting actions that are clearly incorrect, since they are harder to understand and will be mentioned as being unmotivated in the critique regardless of the physician's reason for doing them. Therefore, if there is more than one possible explanatory goal, none of which is relevant, the algorithm does not try to disambiguate the explanation further and the process halts here. Otherwise, the highest ranking (most relevant) non-empty subset of

1. For each action α ordered, TraumaTIQ's plan recognizer extracts from TraumAID's knowledge base a set of *explanatory procedure-goal chains*, PG_α, that could explain the presence of that action:
$$PG_\alpha = \{\langle P \dots G\rangle_1, \dots, \langle P \dots G\rangle_n\}$$
where P is a procedure containing α in its decomposition, and $\langle P \dots G\rangle_i$ is a backward path through the plan graph ending with the goal G.

2. Now consider the set $\Gamma = \{G_i\}$ where G_i is the top level goal ending $\langle P \dots G\rangle_i$. In rank order, Γ consists of Γ_1 the relevant goals, Γ_2 the potentially relevant goals, Γ_3 the previously relevant goals, and Γ_5 all other goals. Let $\Gamma' = \{G_j\}$ be the highest ranking non-empty subset of Γ. If Γ' is the set of irrelevant goals, halt here and add α to the plan with no explanatory procedure-goal chains.

3. Let $\mathcal{P} = \{P_j\}$ where P_j is the procedure that is the child of G_j in PG_α. In rank order, $\mathcal{P}$ consists of: $\mathcal{P}_1$, procedures for which all the actions have been ordered, $\mathcal{P}_2$, procedures for which some of the actions have been ordered, $\mathcal{P}_3$, procedures that are currently in TraumAID's plan, and $\mathcal{P}_4$, all other procedures. Let $\mathcal{P}'$ be the highest ranking non-empty subset of $\mathcal{P}$.

4. Select the paths $PG' \subseteq PG$ such that PG' contains all paths ending with goals in Γ' with children in $\mathcal{P}'$.

5. The paths in PG' are then incorporated into TraumaTIQ's model of the physician's plan, connected to the action α.

Figure 3: The plan recognition algorithm

explanatory goals is selected to be evaluated in phase two.

The second phase considers the procedures in the remaining explanations. These are evaluated according to how strongly the physician's other actions/orders provide additional evidence for them. The more actions in the procedure have been ordered, the more evidence there is in support of the explanation. For simplicity, the procedures are actually sorted according to a four-level scale of evidence:

1. Completed procedures: procedures for which all the actions have been ordered by the physician.

2. Partially completed procedures: procedures for which some of the actions have been ordered.

3. Relevant procedures: procedures that are currently in TraumAID's plan. This means that if an action could address a goal by playing a role in two different procedures, the procedure that TraumAID has selected in its plan is preferred as the explanation for the physician's action.

4. All other procedures.

All procedures in the highest non-empty category are accepted as explanations for the action.

Finally, the explanations with the most relevant top-level goals and the greatest amount of observed evidence are ascribed to the physician and incorporated into TraumaTIQ's model of the physician's plan. Incorporating a new explanation into the plan involves adding new procedures and goals if they are not already present, and adding links between items that are not already connected.

Note that there may be more than one explanation for a given action as long as the explanatory goals are equally relevant and the procedures have the same amount of observed evidence supporting them. For example, in Figure 2 both G_1 and G_2 might be accepted as explanatory goals for the action a_3, provided that both goals are in the same category of relevance and are not irrelevant.

4 Evaluation

The system has been evaluated retrospectively on a database of 97 actual trauma management plans from cases that presented consecutively at the Medical College of Pennsylvania. Out of the 584 actions in these cases, 234 of them were not also part of TraumAID's plan at the time that they were ordered.

Of these 234, 15 could be explained by a goal that TraumAID considered *relevant* to the situation, although it chose to address the goal with a different procedure. Of the remaining 219 actions, 69 could be explained by a goal that was considered to be potentially relevant in the future, given TraumAID's current knowledge about the state of the patient. The plan recognizer failed to explain the remaining 148 actions in terms of relevant or potentially relevant goals.

Part of the reason for these plan recognition failures is that the knowledge base designed for TraumAID's planner is insufficient for the needs of plan recognition. First, many of the actions that TraumaTIQ fails to infer a goal for are broad diagnostic tests that can be used to look for a number of conditions, and the physician may not actually have a specific goal in mind when ordering them. To understand physicians' plans in such cases it is necessary to have a representation of abstract goals that is not currently available in TraumAID 2.0. Since the knowledge base was implemented in support of plan *generation* rather than plan *recognition*, only goals that could be directly operationalized as actions were included.

Second, some goals that physicians may pursue in these cases are not included in TraumAID's knowledge base because its designers opted not to pursue these goals under any circumstances relevant to the current domain of the system. To have a complete plan recognition system, it is necessary to include such goals in the knowledge base.

The urgency inherent in trauma management means that there is limited time available to respond to physicians' actions. It is therefore very important to limit the amount of computation necessary for the plan recognition algorithm. Our plan recognition algorithm is polynomial in the number of ordered actions, and linear in the number of possible explanatory goals per action [3]. Implemented in Lucid Common Lisp and compiled on a Sun 4, it processed 584 actions in an average of 0.023 cpu seconds per action.

5 Conclusion

We believe that the acceptance of clinical decision-support systems for quality assurance depends crucially on their ability to communicate effectively with physicians. To this end, we recognize the importance of reasoning about the complex goals and plans that often arise in medical domains. We have implemented a system that addresses these issues and provides information to physicians in real time about the appropriateness of their goals and how they can best address them.

References

[1] James F. Allen and C. Raymond Perrault. Analyzing intention in utterances. *Artificial Intelligence*, 15:143–178, 1980.

[2] David A. Evans. On the design of medical knowledge-based systems from the perspective of natural-language processing. In *Working notes, AAAI Spring Symposium on AI in Medicine*, pages 49–53, Stanford, CA, 1990.

[3] Abigail S. Gertner. *Critiquing: Effective Decision Support in Time-Critical Domains.* PhD thesis, University of Pennsylvania, Philadelphia, Pennsylvania, November 1995.

[4] Randall W. Hill Jr. and W. Lewis Johnson. Situated plan attribution. *Journal of Artificial Intelligence in Education*, 1995. to appear.

[5] C. P. Langlotz and E. H. Shortliffe. Adapting a consultation system to critique user plans. *International Journal of Man-Machine Studies*, 19:479–496, 1983.

[6] Robert London. Student modeling to support multiple instructional approaches. *User Modeling and User-Adapted Interaction*, 2(1-2):117–154, 1992.

[7] P. L. Miller. *A Critiquing Approach to Expert Computer Advice: ATTENDING.* London: Pittman Press, 1984.

[8] Yuval Shahar and Mark A. Musen. Plan recognition and revision in support of guideline-based care. In *Working notes of the AAAI Spring Symposium on Representing Mental States and Mechanisms*, pages 118–126, Stanford, CA, 1995.

[9] R. L. Teach and E. H. Shortliffe. An analysis of physician attitudes regarding computer-based clinical consultation systems. *Computers and Biomedical Research*, 14:542–558, 1981.

[10] Peter van Beek, Robin Cohen, and Ken Schmidt. From plan critiquing to clarification dialogue for cooperative response generation. *Computational Intelligence*, 9(2):132–154, 1993.

[11] J. van der Lei. *Critiquing Based on Computer-Stored Medical Records.* PhD thesis, Erasmus University, 1991.

[12] B. L. Webber, R. Rymon, and J. R. Clarke. Flexible support for trauma management through goal-directed reasoning and planning. *Artificial Intelligence in Medicine*, 4(2):145–163, April 1992.